CLAIMS

1. A method comprising:

providing a data processing operation involving at least one lookup table, each particular table from said at least one lookup table having a particular lookup table size and a particular lookup table index size; and

creating at least one randomized table in which entries and/or indices are statistically independent from entries and/or indices of said at least one lookup table, each individual table from said at least one randomized table having a randomized table size, wherein a first sum of sizes of all said randomized tables is smaller than a second sum of sizes of all lookup tables, or the maximum index size of said randomized tables is less than the maximum index size of the lookup tables.

2. A method as recited in claim 1, further comprising performing said data processing operation employing said at least one randomized table.

3. A method as recited in claim 1, wherein the step of providing includes obtaining said data processing operation.

4. A method as recited in claim 1, wherein the step of creating said at least one randomized table includes applying a Table Split operation to at least one of said lookup tables resulting in split lookup tables; and/or applying a Table Masking operation to at least one of said lookup tables and/or split lookup tables resulting in masked tables.

5. A method as recited in claim 1, wherein the step of creating said at least one randomized table includes applying a Table Masking operation to at least one of said lookup tables and/or to split lookup tables resulting in masked tables.

6. A method as recited in claim 5, wherein the step of creating said at least one randomized table includes the step of applying a Table Aggregate operation to at least one of said masked tables.

7. A method as recited in claim 4, wherein the step of applying a Table Split operation includes employing a Two-way Byte Table Splitting Method.

8. A method as recited in claim 5, wherein the step of applying a Table Mask operation includes employing a Input-Output Permutation Masking Method.

9. A method as recited in claim 6, wherein the step of applying a Table Aggregate operation includes employing an Entry-wise Algebraic Aggregate Method.

10. A method as recited in claim 1, wherein said at least one table is a table from a COMP128 application.

11. A method as recited in claim 1, wherein a number of elements in said at least one lookup table is given by a power of two.

12. A method as recited in claim 1, further comprising:

employing said at least one randomized table in a cryptographic process;

applying said at least one randomized table for securely handling information in said cryptographic process.

13. A method as recited in claim 12, further comprising:

prior to performing said cryptographic process, transforming the information by applying a secret-sharing operation to the elements of the information where each element of the information is related to multiple elements of the transformed information;

performing the cryptographic process on the transformed information involving the use of said randomized table; and

retransforming the transformed and cryptographically processed information by applying an inverse secret-sharing operation to the transformed and cryptographically processed information.

14. A method as recited in claim 5, wherein indices to at least one masked table of said plurality of masked tables are masked by a single permutation and data values in said at least one masked table are masked by a single permutation.

15. A method as recited in claim 1, further comprising employing the data processing operation as a countermeasure against at least one first-order side-channel attack.

16. A method as recited in claim 5, wherein the step of applying Table Mask operation includes employing permutations for index and/or data values formed by composing several individual permutations.

17. A method as recited in claim 5, wherein the step of applying the Table Mask operation includes employing several individual permutations to defeat at least one higher-order side-channel attack.

18. A method as recited in claim 1, wherein said at least one table is a table from an application of General Countermeasure Against Side-channel Attacks.

19. A method comprising:

providing a lookup table for a data processing operation;

performing a table split operation upon said lookup table in forming a collection of split tables;

performing a table mask operation upon said collection of split tables and/or upon other lookup tables in forming a plurality of masked tables;

performing a table aggregate operation on at least two of said plurality of masked tables in forming at least one aggregate table; and

performing said data processing operation on a combination of split, masked, aggregate and lookup tables.

20. A method comprising:

providing a data processing operation involving at least one lookup table, each particular table from said at least one lookup table having a particular lookup table size and a particular lookup table index size;

declaring any lookup table from said at least one lookup table to be splittable:

if the table lookup size of said any lookup table is larger than an amount of RAM available, or

if the table index size of said any lookup table is larger than available addressing capability;

performing a table split operation upon said any lookup table declared splittable in the step of declaring and forming a collection of split tables;

performing a table mask operation upon said collection of split tables and/or other of said lookup tables forming a plurality of masked tables; and

performing said data processing operation on a combination of split, masked, aggregate and lookup tables.

21. A method as recited in claim 20, further comprising performing at least one table aggregate operation on at least two of said plurality of masked tables forming at least one aggregate table.

22. A method as recited in claim 21, wherein the step of performing said data processing operation includes performing a table aggregate operation whenever a total size of all masked tables exceeds an amount of RAM available.

23. A method as recited in claim 21, wherein the step of providing includes obtaining the data processing operation.

24. A method as recited in Claim 20, wherein the step of performing a table split operation includes employing an Output Divisor Table Splitting Method.

25. A method as recited in Claim 20, wherein the step of performing a table mask operation includes employing an Input-Output XOR Permutation Masking Method.

26. A method as recited in Claim 21, wherein said table aggregate operation includes employing a Byte-wise XOR Aggregate Method.

27. A method as recited in claim 21, further comprising performing said data processing operation on a combination of split, masked, aggregate and lookup tables.

28. A method as recited in claim 20, further comprising employing the data processing operation as a countermeasure against at least one side channel attack.

29. A method as recited in claim 1, wherein a number of elements in said at least one lookup table is 200.

30. An article of manufacture comprising a computer usable medium having computer readable program code means embodied therein for causing resistance to side-channel attacks, the computer readable program code means in said article of manufacture comprising computer readable program code means for causing a computer to effect the steps of claim 1.

31. An article of manufacture comprising a computer usable medium having computer readable program code means embodied therein for causing resistance to side-channel attacks, the computer readable program code means in said article of manufacture comprising computer readable program code means for causing a computer to effect the steps of claim 19.

32. An apparatus comprising:

means for declaring any lookup table from a provided set of lookup table to be splittable if the table lookup size of said any lookup table is larger than an amount of RAM available, or if the table index size of said any lookup table is larger than available addressing capability, each particular table from said set of lookup tables having a particular lookup table size and a particular lookup table index size, said any lookup table;

means for performing a table split operation upon said any lookup table declared splittable in the means for declaring and forming a collection of split tables;

means for performing a table mask operation upon said collection of split tables and/or other of said lookup tables forming a plurality of masked tables; and

means for performing said data processing operation upon a combination of split, masked, aggregate and lookup tables.

33. An apparatus as recited in claim 32, further comprising means for performing a table aggregate operation on at least two of said plurality of masked tables.

34. An apparatus as recited in claim 33, wherein the means for performing a table aggregate operation performs the table aggregate operation when the total size of all masked tables exceeds the amount of RAM available.

35. An apparatus as recited in claim 32, wherein the means for declaring obtains said any lookup tables from another module.

36.. A method comprising

providing a data processing operation involving a first lookup table in a cryptographic process, said lookup table having a first lookup table size;

creating a randomized table in which entries or indices are statistically independent of entries or indices of said first lookup table, said randomized table having a randomized table size being smaller than said first lookup table size;

employing said randomized table for securely handling information in said cryptographic process;

prior to performing the cryptographic process, transforming the information by applying a secret-sharing operation to the elements of the information where each element of the information is related to multiple elements of the transformed information;

performing the cryptographic process on the transformed information involving the use of said randomized table; and

retransforming the transformed and cryptographically processed information by applying an inverse secret-sharing operation to the transformed and cryptographically processed information.

37. A method as recited in claim 36, further comprising performing said data processing operation employing said randomized table.

38. A method as recited in claim 36, wherein said cryptographic process is performed in a cryptographic information processing system or device.

39. A chip card comprising a module implementing the steps of claim 1.

40. A method as recited in claim 1, wherein said at least one lookup table is fixed.

41. An apparatus comprising:

a randomizer module to create at least one randomized table in which entries and/or indices are statistically independent of entries and/or indices of any table from a provided set of lookup tables, each individual table from said at least one randomized table having a randomized table size, wherein:

a first sum of sizes of all said randomized tables is smaller than a second sum of sizes of all said at least one lookup tables, or

the maximum index size of said randomized tables is less than the maximum index size of the lookup tables; and

a processing module to perform said data processing operation employing said first randomized table.

42. An apparatus as recited in claim 41, wherein the randomizer module forms said provided set of lookup tables.

43. An apparatus as recited in claim 41, wherein the randomizer module includes a splitting module to perform a Table Split operation upon at least a subset of said set of lookup tables resulting in split lookup tables.

44. An apparatus as recited in claim 41, wherein the randomizer module includes a masking module to perform a Table Masking operation upon at least a subset of said set of lookup tables and/or split lookup tables forming a set of masked tables.

45. An apparatus as recited in claim 43, wherein the randomizing module includes an aggregating module to perform a Table Aggregate operation to at least one masked table.

46. An apparatus as recited in claim 43, wherein the splitting module includes an Unequal Table Splitter Module which applies the Unequal Table Split Method for performing a Table Split Operation.

47. An apparatus as recited in claim 44, wherein the masking module includes an Input-Output XOR Permutation Masking module which applies the Input-Output XOR Permutation Masking Method for performing a Table Mask Operation.

48. An apparatus as recited in claim 45, wherein the aggregating module includes an Byte-wise XOR Aggregating Module which applies the Byte-wise XOR Aggregating Method for performing a Table Aggregate Operation.

49. A computer program product comprising a computer usable medium having computer readable program code means embodied therein for causing resistance to side-channel attacks, the computer readable program code means in said computer program product comprising

computer readable program code means for causing a computer to effect the functions of claim 32.

50. An apparatus comprising

a splitting module to perform a table split operation upon a provided set of lookup tables, forming a plurality of split tables;

a masking module to perform a table mask operation upon said collection of split tables and/or other lookup tables forming at least one masked tables;

an aggregating module to perform a table aggregate operation on a subset of said plurality of masked tables, forming at least one aggregate tables; and

a processing module to perform a data processing operation employing a combination of split, masked, aggregate and lookup tables.

51. An article of manufacture comprising a computer usable medium having computer readable program code means embodied therein for causing resistance to side-channel attacks, the computer readable program code means in said article of manufacture comprising computer readable program code means for causing a computer to effect the steps of claim 20.

52. An article of manufacture comprising a computer usable medium having computer readable program code means embodied therein for causing resistance to side-channel attacks, the computer readable program code means in said article of manufacture comprising computer readable program code means for causing a computer to effect the steps of claim 36.

53. A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for causing resistance to side-channel attacks, said method steps comprising the steps of claim 1.

54. A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for causing resistance to side-channel attacks, said method steps comprising the steps of claim 20.

55. A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for causing resistance to side-channel attacks, said method steps comprising the steps of claim 36.

56. A computer program product comprising a computer usable medium having computer readable program code means embodied therein for causing resistance to side-channel attacks, the computer readable program code means in said computer program product comprising computer readable program code means for causing a computer to effect the functions of claim 41.

57. A computer program product comprising a computer usable medium having computer readable program code means embodied therein for causing resistance to side-channel attacks, the computer readable program code means in said computer program product comprising computer readable program code means for causing a computer to effect the functions of claim 50.